UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/825,466	04/15/2004	Timothy Nephi Tillotson	10030536-1	8111

22878 7590 05/11/2009

AGILENT TECHNOLOGIES INC.
INTELLECTUAL PROPERTY ADMINISTRATION,LEGAL DEPT.
MS BLDG. E P.O. BOX 7599
LOVELAND, CO 80537

EXAMINER	
YU, HENRY W	

ART UNIT	PAPER NUMBER
2182	

NOTIFICATION DATE	DELIVERY MODE
05/11/2009	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

IPOPS.LEGAL@agilent.com

Office Action Summary	**Application No.** 10/825,466	**Applicant(s)** TILLOTSON ET AL.
	Examiner HENRY YU	**Art Unit** 2182

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *02 March 2009*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-15 and 17-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-15 and 17-20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *15 April 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

INFORMATION CONCERNING RESPONSES

Response to Amendment

1. This Office Action is in response to applicant's communication filed on March 2, 2009, in response to PTO Office Action mailed on December 1, 2008. The Applicants' remarks and amendments to the claims and/or the specification were considered with the results that follow.

2. In response to the last Office Action, **claims 1, 10, and 15** have been amended. **Claim 16** has been cancelled. As a result, **claims 1-15 and 17-20** are now pending in this application.

Response to Arguments

3. Applicants' arguments filed on March 2, 2009, in response to PTO Office Action mailed on December 1, 2008, have been fully considered but are not persuasive.

Applicants have argued that nothing in Fuller, III et al. (Patent Number US 7,134,081 B2) or Robison et al. (Publication Number US 2005/0060693 A1) discloses receiving at an instrument a command and converting the command from a first (SCPI) protocol to a second (.NET) protocol. Again, Examiner points out that Fuller, III et al. discloses the idea of converting one form of data (the idea can also be applied to commands and instructions) to another within the passage in **[Column 5]**, particularly after parsing responses **[Column 5, lines 7-12]** the resulting tokens (done after parsing as noted in **[Column 5, lines 13-22]** can then be used to create text-based code which include the .NET format **[Column 5, lines 28-30]**).

Concerning Robison et al. and the argument that Robinson et al. does not disclose the use of protocols, from the Examiner's understanding protocols pertain to a set of rules and formats for communication. In the case of Robinson et al., one can see that there exists the use of "protocols" in the sense that one such protocol utilizes character strings while another protocol utilizes objects **[Page 1, paragraph 0017]**. What Robinson et al. disclose is a means of converting between the two before passing the parameters and commands to an action handler. Examiner has further clarified why it would be obvious to one of ordinary skill in the art to modify Fuller, III et al. with elements of Robinson et al. (see section entitled **REJECTIONS BASED ON PRIOR ART**).

Examiner notes that Applicants has amended **claim 15** to include the use of a format converter that converts one stream format into another. Upon further review, Examiner notes that Robinson et al. discloses a similar system where each successive token is compared **[Page 2, paragraph 0018]**. The use of the word "successive" indicates a streaming manner in that data/parameters are transferred in sequence.

REJECTIONS BASED ON PRIOR ART

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. **Claims 1, 3-4, 8, 10-11, 15, and 16-18** are rejected under 35 U.S.C. 103(a) as being unpatentable over Fuller, III et al. (Patent Number US 7,134,081 B2) in view of Robison et al. (Publication Number US 2005/0060693 A1).

As per **claim 1**, Fuller, III et al. discloses an instrument controlling system that handles responses **[Abstract]**, as well as the use of SCPI **[Column 4, lines 55-57]** and .NET **[Column 5, lines 29-35]**. While Fuller, III et al. discloses "*receiving at an instrument via a communication link a communication from a client comprising a processor and a memory, the communication comprising one of an SCPI protocol command and SCPI protocol query* ***(a command is to be sent to a selected instrument, which can conform to the SCPI standard; Column 4, lines 49-57 and 63-65)***," Fuller, III et al. does not explicitly disclose the idea of command translation in the limitations "*a method for translating a communication between Standard Commands for Programmable Instrumentation (SCPI) protocol and .NET protocol, the method comprising: when the communication is the SCPI protocol command, converting the SCPI protocol command to a .NET protocol command,*" "*evaluating the .NET protocol command to determine the validity of parameters sent from the client with the SCPI protocol command,*" "*otherwise, when the communication is the SCPI protocol query, converting the SCPI protocol query to a .NET protocol query,*" "*evaluating the .NET protocol query to determine the validity of parameters sent from the client with the SCPI protocol query,*" or "*calling an appropriate Application Program Interface (API) of an instrument application in the instrument, wherein the communication is intended for the*

instrument application and wherein the API is responsive to method calls in the .NET protocol."

Robison et al. discloses the idea of command translation in the limitations "*a method for translating between Standard Commands for Programmable Instrumentation (SCPI) protocol* ***(for this part of the rejection, labeled the 'first' protocol)*** *and .NET* ***(for this part of the rejection, labeled the 'second' protocol)*** *protocol communications, comprising: when the communication is the* ***(first)*** *protocol command, converting the…protocol command to a* ***(second)*** *protocol command* ***(the command string is syntactically matched by the command processor code portion and all parameter values within the command string are converted to their corresponding data object; Page 2, paragraph 0022)***" and "*evaluating the* ***(second)*** *protocol command to determine the validity of parameters sent from the client with the* ***(first)*** *protocol command* ***(the corresponding data objects can then be further validated; Page 2, paragraph 0022)***." The same also applies to "*otherwise, when the communication is the SCPI protocol query, converting the SCPI protocol query to a .NET protocol query*" and "*evaluating the .NET protocol query to determine the validity of parameters sent from the client with the SCPI protocol query*," where a query can also act as a particular command subset that instructs a device/system to return a specific value (as noted by Fuller, III et al. **[Column 4, line 49]**).

Robison et al. discloses "*calling an appropriate Application Program Interface (API) of an instrument application in the instrument, wherein the communication is intended for the instrument application and wherein the API is responsive to method*

calls in the…protocol ***(action handler code is invoked to actually execute the task that corresponds to the command; Page 2, paragraph 0022)***."

Fuller, III et al. and Robison et al. are analogous art in that they are from the same field of command processing.

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the method and system disclosed by Fuller, III et al. to include a means of translating and evaluating commands from one protocol to another as disclosed by Robison et al., which notes the need to create meaningful, actionable objected or data structures **[Page 1, paragraph 0003]**. Robison et al. also notes that there are prior art systems where the syntax of all commands is hard-coded into the parser **[Page 1, paragraph 0004]**, and hence it would be necessary to translate commands protocols to accommodate such systems. Furthermore, a system that is capable of translating between protocols has greater flexibility in its ability to communicate among many different types of systems/devices, and by validating these translations one can be assured that the translated commands will be processed as intended.

As per **claims 3 and 10**, the combination of Fuller, III et al. and Robison et al. discloses "*the method*" (*see rejection to* ***claims 1 and 8***). Fuller, III et al. further discloses "*when the SCPI protocol query or the SCPI protocol command requires response from the instrument application, forming a .NET protocol response message to the communication* ***(a command is sent to the instrument (step 304), with a response from the instrument then being received (step 306); FIG. 9)***."

While Robison et al. discloses the idea of command translation **[Page 2, paragraph 0022]**, Fuller, III et al. discloses the handling of responses in the limitations "*translating the (.NET) protocol response message to a SCPI protocol response message* ***(steps 306 to 310, where a response from the instrument is received, with the instrument response parsed and code created; FIG. 9)****, wherein the SCPI protocol response message comprises contents of nodes of a SCPI hierarchical tree structure* ***(the instrument responses can be pre-parsed, which require the use of a library of data import functions to pre-parse instrument responses (Column 15, lines 22-25). It should be noted that the idea of a hieratical tree is implied through Column 14, lines 47-52, where certain options may be only exposed when necessary, particularly though an 'Advanced Features' tab (which is a step beyond basic functions))***" and "*transferring the SCPI protocol response message to the client* ***(a command is sent to the instrument (step 304), with a response from the instrument then being received (step 306); FIG. 9)***." **Claim 10** is rejected in a similar fashion.

As per **claims 4 and 11**, the combination of Fuller, III et al. and Robison et al. discloses "*the method*" (*see rejection to* ***claims 1 and 8***). Robison et al. further discloses "*before transferring the (SCPI) protocol response message to the client, converting the (SCPI) protocol response message to (SCPI) format order* ***(all parameter values within the command string are to be converted to their corresponding data objects before any action handler code is invoked (Page 2, paragraph 0022). This passage demonstrates the need to convert data object***

types to the appropriate types before being processed (similar to the instant application's need to convert the protocol response message to the appropriate format before being sent to the client system))." **Claim 11** is rejected in a similar fashion.

As per **claim 8**, Fuller, III et al. discloses "*a computer readable memory device embodying a computer program of instructions executable by a computer* ***(a system and method for querying message-based instruments, automatically…parsing the responses, and generating code that encapsulates the connection/communication with the instrument and the parsing of the response; Column 4, lines 30-35)***." However, Fuller, III et al. does not disclose "*the instructions comprising: when a communication is a SCPI protocol command from a client, converting the SCPI protocol command to a .NET protocol command*," "*evaluating the .NET protocol command to determine the validity of parameters sent from the client with the SCPI protocol command*," "*otherwise, when the communication is a SCPI protocol query from the client, converting the SCPI protocol query to a .NET protocol query*," "*evaluating the .NET protocol query to determine the validity of parameters sent from the client with the SCPI protocol query*," or "*calling an appropriate Application Program Interface (API) of an instrument application, wherein the communication is intended for the instrument application and wherein the API is responsive to method calls in the .NET protocol*."

Robison et al. discloses "*the instructions comprising: when the communication is a SCPI* ***(for this rejection, the protocol is noted as a 'first' protocol)*** *protocol*

command from a client, converting the SCPI protocol command to a .NET ***(for this rejection, the protocol is labeled as a 'second' protocol)*** *protocol command* ***(the command string is syntactically matched by the command processor code portion and all parameter values within the command string are converted to their corresponding data object; Page 2, paragraph 0022)***" and "*evaluating the .NET* ***(second)*** *protocol command to determine the validity of parameters sent from the client with the SCPI* ***(first)*** *protocol command* ***(the corresponding data objects can then be further validated; Page 2, paragraph 0022)***." The same also applies to "*otherwise, when the communication is a SCPI protocol query from the client, converting the SCPI protocol query to a .NET protocol query*" and "*evaluating the .NET protocol query to determine the validity of parameters sent from the client with the SCPI protocol query*," where a query can also act as a particular command subset that instructs a device/system to return a specific values (as noted by Fuller, III et al. **[Column 4, lines 49]**). Robison et al. also discloses "*calling an appropriate Application Program Interface (API) of an instrument application, wherein the communication is intended for the instrument application and wherein the API is responsive to method calls in the .NET protocol* ***(action handler code is invoked to actually execute the task that corresponding to the command; Page 2 paragraph 0022)***."

Fuller, III et al. and Robison et al. are analogous art in that they are from the same field of command processing.

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the computer readable memory device embodying a computer

program of instructions executable by the computer as disclosed by Fuller, III et al. to include a means of translating commands from one protocol to another as disclosed by Robison et al., which notes the need to create meaningful, actionable objected or data structures **[Page 1, paragraph 0003]**. Robison et al. also notes that there are prior art systems where the syntax of all commands is hard-coded into the parser **[Page 1, paragraph 0004]**, and hence it would be necessary to translate commands protocols to accommodate such systems. Furthermore, a system that is capable of translating between protocols has greater flexibility in its ability to communicate among many different types of systems/devices, and by validating these translations one can be assured that the translated commands will be processed as intended.

As per **claim 15**, Fuller, III et al. discloses "*a system, comprising: a format converter configured to receive a Standard Commands for Programmable Instrumentation (SCPI) protocol communication from a client* ***(the system contains a means of parsing responses from an instrument (steps 308 to 310 in FIG. 9). Since the method is used to determine correct token characteristics (step 364 in FIG. 10), the same method can be applied in both directions to ensure proper command processing. Fuller, III et al. also discloses the use of parsing in Column 4, lines 51-57 and Column 5, lines 1-12)*** *via a communication link and to convert a...format of the SCPI protocol communication into a .NET...format* ***(the result of the graphical token specification may be used to automatically create text-based code, such as .NET; Column 5, lines 29-31).*** " Though Fuller, III et al. discloses the use of SCPI and .NET protocols, Fuller, III et al. does not explicitly disclose the use of

an evaluator or similar module as disclosed in the limitation "*a parser configured to translate commands and queries of the communication having the .NET stream format from the SCPI protocol into a .NET protocol,*" "*an evaluator module, configured to evaluate the .NET protocol commands and queries to determine the validity of parameters sent from the client with the SCPI protocol communication,*" nor explicitly the idea of streaming communication.

Robison et al. not only explicitly discloses a component that parses commands in the passage **[a command processor receives a command-string that is parsed into character string tokens; Page 2, paragraph 0018]**. Robison et al. also discloses "*an evaluator module* ***(command processor code portion)****, configured to evaluate the .NET protocol commands and queries to determine the validity of parameters sent from the client with the SCPI protocol communication* ***()(the corresponding data objects can then be further validated; Page 2, paragraph 0022)***." Robison et al. further discloses the idea of streaming communications and translations **[each successive token is compared (Page 2, paragraph 0018). The use of the word 'successive' indicates a streaming manner in that data/parametrs are transferred in sequence]**.

Fuller, III et al. and Robison et al. are analogous art in that they are from the same field of command processing.

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the system as disclosed by Fuller, III et al. to include a means of translating commands from one protocol to another as disclosed by Robison et al., which notes the need to create meaningful, actionable objected or data structures

[Page 1, paragraph 0003]. Robison et al. also notes that there are prior art systems where the syntax of all commands is hard-coded into the parser **[Page 1, paragraph 0004]**, and hence it would be necessary to translate commands protocols to accommodate such systems. Furthermore, a system that is capable of translating between protocols has greater flexibility in its ability to communicate among many different types of systems/devices, and by validating these translations one can be assured that the translated commands will be processed as intended.

As per **claim 17**, the combination of Fuller, III et al. and Robison et al. discloses "*the system*" (*see rejection to* ***claim 15*** above). While Robison et al. discloses the idea of translation as **[the command string is syntactically matched by the command processor code portion and all parameter values within the command string are converted to their corresponding data object (Page 2, paragraph 0022)]**, Fuller, III et al. discloses handling responses as noted in the limitations "*a first translator module configured to translate a .NET response from an instrument application to a SCPI protocol response* ***(the system receives a response from the instrument and parses the instrument response; FIG. 9; Column 15, lines 1-7; Column 20, lines 30-36)***."

As per **claim 18**, the combination of Fuller, III et al. and Robison et al. discloses "*the system*" (*see rejection to* ***claim 15*** above). While Robison et al. discloses the idea of translation as **[the command string is syntactically matched by the command processor code portion and all parameter values within the command string are converted to their corresponding data object (Page 2, paragraph 0022)]**, Fuller, III et al. discloses handling responses as noted in the limitations discloses "*a second*

format converter module configured to convert the SCPI protocol response in a .NET stream format into SCPI format order ***(the system receives a response from the instrument and parses the instrument response; FIG. 9; Column 15, lines 1-7; Column 20, lines 30-36)***."

6. **Claims 2 and 9** are rejected under 35 U.S.C. 103(a) as being unpatentable over Fuller, III et al. (Patent Number US 7,134,081 B2) in view of Robison et al. (Publication Number US 2005/0060693 A1) and in further view of Durian et al. (Publication Number US 2002/0025832 A1).

As per **claims 2 and 9**, the combination of Fuller, III et al. and Robison et al. discloses "*the method*" (*see rejection to* ***claims 1 and 8*** *above*). However, the combination of Fuller, III et al. and Robison et al. does not explicitly disclose "*before converting the SCPI protocol command to the .NET protocol command, placing the SCPI protocol command into .NET stream format*" or "*before the method step converting the SCPI protocol query to the .NET protocol query, placing the SCPI protocol command into .NET stream format*."

Durian et al. discloses "*before converting the SCPI protocol command to the .NET protocol command* ***(translating communications channel control commands within a wireless communication system)****, placing the SCPI protocol command into .NET stream format* ***(communications are generally translated into a format required by the receiving device (in this case a telephone); Page 17, paragraph 0121)***." Durian et al. also discloses "*before the method step converting the SCPI protocol query to the .NET protocol query, placing the SCPI protocol command into*

*.NET stream format **(there is a situation where one type of formatting is done before another. In this case, communications channel control commands are translated into at least a first command selected from a set of system commands before being translated into corresponding wireless communication device control commands that can be understood by the receiving device; Pages 17-18, paragraph 0121)**.*"

Fuller, III et al., Robison et al., and Durian et al. are analogous art in that they are from the same field of command processing.

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the method/system as disclosed by the combination of Fuller, III et al. and Robison et al. to include as translation into a communication stream disclosed by Durian et al., which notes the desirability of having a plurality of different devices or applications to interconnect with a communications device as well as to communicate over a channel established by the by the communications device at substantially the same time **[Pages 1-2, paragraph 0010]**. **Claim 9** is rejected in similar fashion.

7. **Claims 5-7, 12-13, and 19** are rejected under 35 U.S.C. 103(a) as being unpatentable over Fuller, III et al. (Patent Number US 7,134,081 B2) in view of Robison et al. (Publication Number US 2005/0060693 A1) and in further view of Hall et al. (Patent Number US 5,974,541).

As per **claims 5 and 12**, the combination of Fuller, III et al. and Robison et al. discloses "*the method*" (*see rejection to **claims 1 and 8** above*). Though Fuller, III et al. discloses the use of .NET **[Column 5, lines 29-35]** and Robison et al. discloses the

idea of command translation **[the command string is syntactically matched by the command processor code portion and all parameter values within the command string are converted to their corresponding data object; Page 2, paragraph 0022]**, which applies to the limitation "*converting the out of band signal IEEE 488.1 protocol signal to a .NET event*," the combination of Fuller, III et al. and Robison et al. does not explicitly disclose "*asynchronously receiving an out of band IEEE 488.1 protocol signal from the client*," "*converting the out of band signal IEEE 488.1 protocol signal to a .NET event*," or "*transferring the out of band signal IEEE 488.1 protocol signal to the instrument application*."

Hall et al. discloses the use of IEEE 488.1, which applies to the limitation "*converting the out of band signal IEEE 488.1 protocol signal* ***(the system utilizes GPIB, also known as the IEEE 388.1-1987; Column 1, lines 66-67)*** *to a .NET event*." Hall et al. also discloses "*asynchronously receiving an out of band IEEE 488.1* ***(the system utilizes GPIB, also known as the IEEE 488.1-1987; Column 1, lines 66-67)*** *protocol signal from the client* ***(receiving a notify request from the GPIB application (step 302); FIG. 3)***" and "*transferring the out of band signal IEEE 488.1 protocol signal to the instrument application* ***(anything sent from the GPIB application ends up at the GPIB hardware (FIG. 2). The process can be done asynchronously; Column 2, lines 38-40)***."

Fuller, III et al., Robison et al., and Hall et al. are analogous art in that they are from the same field of command processing.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the method disclosed by the combination of Fuller, III et al. and Robison et al. to include a means of processing IEEE 488.1 asynchronously as disclosed by Hall et al., which notes the desirability of asynchronous processing compared to prior art systems, where the GPIB driver level software has traditionally been required to poll a device to determine when an event occurs. However, such a process typically requires a large amount of unnecessary processor time, thus consuming valuable CPU resources **[Column 2, lines 29-35]**. **Claim 12** is rejected in a similar fashion.

As per **claims 6 and 13**, the combination of Fuller, III et al. and Robison et al. discloses "*the method*" (*see rejection to* ***claims 1 and 8*** *above*). However, the combination of Fuller, III et al. and Robison et al. does not disclose "*when an event occurs in the instrument application, posting a notice of event occurrence in a status module*" or "*asynchronously notifying the client of event occurrence*."

Hall et al. discloses "*when an event occurs in the instrument application, posting a notice of event occurrence in a status module* ***(the method operates to asynchronously notify a user when one or more GPIB events occur in the GPIB system; Column 2, lines 40-42)***" and "*asynchronously notifying the client of event occurrence* ***(represented by step 302 of receiving a notify request from the GPIB application; FIG. 3)***."

Fuller, III et al., Robison et al., and Hall et al. are analogous art in that they are from the same field of command processing.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the method as disclosed by the combination of Fuller, III et al. and Robison et al. to include a means of processing IEEE 488.1 asynchronously as disclosed by Hall et al., which notes the desirability of asynchronous processing compared to prior art systems, where the GPIB driver level software has traditionally been required to poll a device to determine when an event occurs. However, such a process typically requires a large amount of unnecessary processor time, thus consuming valuable CPU resources **[Column 2, lines 29-35]**. **<u>Claim 13</u>** is rejected in similar fashion.

As per **<u>claim 7</u>**, the combination of Fuller, III et al., Robison et al., and Hall et al. discloses "*the method*" (*see rejection to **<u>claim 6 and 13</u>** above*). Fuller, III et al. further discloses "*forming a .NET protocol response message to the query* ***(a command is sent to the instrument (step 304), with a response from the instrument then being received (step 306); FIG. 9)***" while Robison et al. further discloses "*translating the .NET protocol response message to a SCPI protocol response message* ***(the command string is syntactically matched by the command processor code portion and all parameter values within the command string are converted to their corresponding data object; Page 2, paragraph 0022)***" and "*transferring the SCPI protocol response message to the client* ***(all parameter values within the command string are to be converted to their corresponding data objects before any action handler code is invoked; Page 2, paragraph 0022)***."

Hall et al. further discloses "*after asynchronously notifying the client of event occurrence* ***(represented by step 302 of receiving a notify request from the GPIB application; FIG. 3)****, receiving a query from the client requesting detailed information regarding the event occurrence* ***(the system has a means of ascertaining more information regarding the event occurrence including: monitoring events specified by the event information (step 304) and determining that an event specified by the information has occurred (step 306) before invoking a callback function in response to the event (step 308); FIG. 3)***." **Claim 14** is rejected in similar fashion.

As per **claim 19**, the combination of Fuller, III et al. and Robison et al. discloses "*the system*" (*see rejection to* ***claim 15*** *above*). Though Robison et al. discloses the idea of command translation as disclosed in "*a third format converter module configured to convert an out of band IEEE 488.1 signal into a .NET signal* ***(the command string is syntactically matched by the command processor code portion and all parameter values within the command string are converted to their corresponding data object; Page 2, paragraph 0022)***," the combination of Fuller, III et al. and Robison et al. does not explicitly disclose "*an out of band IEEE 488.1 signal*," which is disclosed by Hall et al. as **[the system utilizes GPIB, also known as the IEEE 488.1-1987 (Column 1, lines 66-67)]** and **[anything sent from the GPIB application ends up at the GPIB hardware (FIG. 2). The process can be done asynchronously (Column 2, lines 38-40)]**.

Fuller, III et al., Robison et al., and Hall et al. are analogous art in that they are from the same field of command processing.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the system as disclosed by the combination of Fuller, III et al. and Robison et al. to include a means of processing IEEE 488.1 asynchronously as disclosed by Hall et al., which notes the desirability of asynchronous processing compared to prior art systems, where the GPIB driver level software has traditionally been required to poll a device to determine when an event occurs. However, such a process typically requires a large amount of unnecessary processor time, thus consuming valuable CPU resources **[Column 2, lines 29-35]**.

8. **<u>Claim 20</u>** is rejected under 35 U.S.C. 103(a) as being unpatentable over Fuller, III et al. (Patent Number US 7,134,081 B2) in view of Robison et al. (Publication Number US 2005/0060693 A1) and in further view of Dobson et al. (Patent Number US 6,766,386 B2).

As per **<u>claim 20</u>**, the combination of Fuller, III et al. and Robison et al. discloses "*the system*" (*see rejection to **<u>claim 15</u>** above*). Though Robison et al. discloses the idea of command translation as disclosed in "*an event translator module configured to receive notice of event occurrence from the status module and to translate that notice into a SCPI status notification* ***(the command string is syntactically matched by the command processor code portion and all parameter values within the command string are converted to their corresponding data object; Page 2, paragraph 0022)***," the combination of Fuller, III et al. and Robison et al. does not explicitly disclose "*a*

status module comprising an event message queue and a status register wherein the event message queue and the status register store event occurrence information from an instrument application."

Dobson et al. discloses "*a status module comprising an event message queue* ***(FIFO write control 414 in conjunction with a read data memory 416 in a first-in-first-out fashion; Column 6, lines 65-67; Column 7, lines 1-5)*** *and a status register* ***(status register 418; FIG. 4)*** *wherein the event message queue and the status register store event occurrence information from an instrument application* ***(Column 8, lines 25-37)***."

Fuller, III et al., Robison et al., and Dobson et al. are analogous art in that they are from the same field of command processing.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the method as disclosed by the combination of Fuller, III et al. and Robison et al. to include a FIFO memory and status register as disclosed by Dobson et al. in order to prevent potential latencies and delays **[Column 1, lines 18-24]**.

CLOSING COMMENTS

Conclusions

9. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to HENRY YU whose telephone number is (571)272-9779. The examiner can normally be reached on Monday to Friday, 8:00 AM to 5:30 PM EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, TARIQ HAFIZ can be reached on (571) 272-6729. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/H. Y./
Examiner, Art Unit 2182

/Tammara Peyton/
Primary Examiner, 2182
April 30, 2009